Information Clash Theory

This theory proposes the conflict between two residing information within the internet that holds the same purpose. These conflicts are due to massive quantities of information that we are possibly experiencing today and might be in the future. The quantities of information within the internet will reach the probability needed for multiple information's answers to differ to one another.

The evidence of the possibility of this theorem is the difference in context from two colliding sources that share the same purpose. Since the amount of information is increasing within the internet, the chances of having two sources or information that have a conflicting answer will also increase. An example of this is two different interpretations of equations that have the same purpose to answer and represent a particular process also as a solution but contains a difference in their properties or answers.

Simply, the rising technology and capacity of every database will result in a more possible increase in the amount of information which will have its conflicts in terms of value, answer, and interpretation. More varied sources will come out that will put in conflict from specific information that could cause confusion towards users. Information Clash Theory could be present but uncertain in today's situation since there is no present technology that could keep track of every information's value, answer, and interpretation.

Information Overload Theory

Information Overload Theory is a hypothetical possibility wherein data, media, and information quantities will reach its limit via the internet or a similar process that will make the data, media, and information to become indeterminable. It is related to Borough's Data Translocation where

particular information, media, and data that are needed cannot be retrieved due to the amount of information, media, and data. This theory refers to the whole situation including the irretrievable information, media, and data by a specific means caused by massive quantities that will result in the information uncertainty principle.

It could be a problem if our future will have inconvenient access and finding resources due to the amount of information. The same instance applies when a server that reaches its limits of clients gets affected by dysfunctions that are related to the server.

Information Uncertainty Principle is when users had conflicts determining an answer, process, and value due to their difference in resources, equations, and information. Possibly, all information will have its corresponding digital equivalence and from the contents of different authors, historians, and mathematicians, there will be a problem in credibility and determining the right value, meaning, and answer.

In Philosophy, there is no such thing as an absolute fact so we cope and develop varying branches that express standards that are expressed by a spectrum from which information we can trust. We develop laws, principles, measurements, and terms such as scientific facts so that we can make sense of anything. It is developed throughout centuries and due to new developments, we are able to make sense some of this world. However, any information will emerge and not all can't be classified. Humans will have problems determining which must be used and credible. This problem is being classified by programs that are made by individuals but the scopes of programming are only limited. The possible method for this hypothetical problem would be the use of Artificial Intelligence or a program that have larger scopes on determining the meaning as well as answers to sort this information.

A computer program is a collection of instructions that performs a specific task when executed by a computer. A computer requires programs to function. A computer program is usually written by a computer programmer in a programming language. From the program in its human-readable form of the source code, a compiler can derive machine code—a form consisting of instructions that the computer can directly execute. Alternatively, a computer program may be executed with the aid of an interpreter. A collection of computer programs, libraries, and related data are referred to as software. Computer programs may be categorized along functional lines, such as application software and system software. The underlying method used for some calculation or manipulation is known as an algorithm.

Computer programming is a way of giving computers instructions about what they should do next. These instructions are known as code, and computer programmers write code to solve problems or perform a task. The end goal is to create something: that could mean anything from a web page, or a piece of software, or even just a pretty picture. That's why computer programming is often described as a mix of art and science; it's technical and analytical, yet creative at the same time.

A program does nothing more than telling the computer how to accept some type of input, manipulate that input, and spit it back out again in some form that humans find useful. Table 1 lists some common types of programs, the types of input that they accept, and the output that they produce.

Essentially, a program tells the computer how to solve a specific problem. Because the world is full of problems, the number and variety of programs that people can write for computers is practically endless. But to tell a computer how to solve one big problem, you usually must tell the computer how to solve a bunch of little problems that make up the bigger problem.

Here's the future of programming according to Yoppworks

8 Predictions of Future Programming Based on 2017 Trends

Computer programming involves writing instructions that a computer system follows, or writing and testing the code for software and applications. In accomplishing these tasks, programmers must learn languages such as C++, Java, or Scala. Some of the job responsibilities are similar to those of software developers. They include modeling and planning the way in which code will be written.

Programmers often work with integrated development environments and are involved in building, fixing, and modifying:

1.Computer operating systems

2.Social networking applications

3.Software-as-a-Service solutions

4.Automated internet search platforms

5.Artificial intelligence programs

6.Voice recognition systems

Scala Will Become More Mainstream

Released in 2003, after design efforts began in 2001, Scalable Language or Scala is an object-oriented programming language that is interoperable with Java. A program written in this language will run in any environment that Java runs in, making it highly versatile. The similarity

makes it attractive to programmers. Capabilities include support for functional programming (which will be highlighted next) and compatibility with algebraic and anonymous data types.

Operator overloading, raw strings, and named parameters are available, although Scala can run on a Java Virtual Machine that doesn't include these functions. It supports cluster computing as well. Yoppworks offers classes and training on building applications using this language with Akka, an open-source toolkit. It enables developers to create software applications on their computer, so long as they have access to the internet.

Applications developed in this manner are efficient and easily scalable, further boosting the appeal and potential of Scalable Language.

Functional Programming

This method of programming uses declarative statements and expressions, focusing on mathematical functions to define statements. It is based somewhat on lambda calculus. Despite being rooted in a system developed in the 1930s, the concept is now looking like a fundamental one for writing the code of the future.

Functional Type Programming Will Be the Basis for Writing JavaScript

However, it won't be limited to HTM, imperative object-oriented, or other singular aspects of code writing, but for drafting entire sets of code. Some elements from the past are no longer necessary and many, in fact, have become irrelevant. Tools for creating more legible code are available that work with the advanced computing power of today. Programmers can write shorter, less complex code for higher-order tasks and by combining fundamental functions.

Programmers Will Become More Dependent on Functional Languages

The concept supports running sections of software in parallel, across different machines and CPU cores. That eliminates the need for complex synchronization. Therefore, Web requests and other functions requiring concurrent processing can be better managed. The trends also affect programmers of smartphone applications, interconnected devices, and servers that support the interactions between them.

Languages such as Scala, Haskell, and Clojure have not only grown in popularity but are expected to continue as the demands for flexible, functional programming expand.

Open Source

Open source software has already dominated the computer programming world. Vast networks of professionals and other enthusiasts have collaborated to make application solutions better. One can access code and modify, copy, and share it. The Apache HTTP Server, Firefox, MySQL, and Perl are all examples of open source programs.\

Apache Spark Will Dominate Data Processing

Datasets are becoming increasingly larger, but this open source framework uses fault tolerance and data parallelism to allow programming on a different level. It provides an interface for programming data clusters. Built by software developers from more than 200 companies, Apache Spark's features are supportive of big data processing.

It runs on any platform and is speedier than other solutions; in fact, it is 100 times as fast as Hadoop clusters. The same data can be used to execute multiple jobs, thanks to in-memory data sharing and cyclic data flow, demonstrating the platform's potential as the demand for faster data processing accelerates.

Open Source Won't Only Be an Interest to Programmers

Programmers love to tinker with open source, but they and people taking computer programming courses at Yoppworks won't be the only ones interested in the concept. Already, we are all using open source software when we visit websites, stream music, chat with friends, and check email. It's used to route and transmit data, right down to the core of today's communication infrastructure, which is based on remote computers, mobile phone applications, and Web browsers. Remote computing, or cloud computing, is increasing in importance and is sure to continue as we rely more on internet-connected devices.

The Preference for Open Source Software Will Be Ubiquitous

The concept appeals to many different types of people. It gives them control over the code. Even non-programmers can make use of the software because they can tweak it to their needs, rather than use it as marketing. The appeal is also growing because such software is easier to study than other types; for example, students can share their work while developing their skills. Open source has changed the way people learn computer programming, right down to sharing mistakes to help others avoid them.

Fixes, updates, and upgrades can be made without the permission of original programmers. Quick fixes can mean the software is more secure. Also, since programmers tend to work on open source software over the long term, they can depend on these tools without them becoming obsolete. Therefore, reliance on open standards may mean more product stability than with many proprietary software tools.

Big Data

Scala, the programming language used to write Apache Spark, and which is used for building applications with Akka, is part of this movement. Aside from fueling the demand for related programming courses, this has even more potential impacts on the future of programming. These points below highlight both the future and what is happening now.

Big Data Adoption Is No Longer Limited to Analytics

Not too long ago, IT departments were focused on using big data for analyzing network activity, usage, and security, among other things. Today, the applications are increasingly driving the demand in all industries. Many analytical processes will still require vast quantities of data, but big data's application in banking and securities, communications and entertainment, healthcare, education, manufacturing, retail, government, energy, insurance, and transportation give it leverage that computer programmers cannot ignore.

The programming of the future is sure to be different than that of the past, in many ways, and is being shaped by the trends of today. From new languages to functional programming and the use of big data, there are bold predictions swirling about the field of computer programming. Here are eight of them that paint a detailed picture of what may be to come.

Big Data Is Fueling the Potential for AI and Machine Learning

There is a connection with the emergence of data virtualization, as big data analytics are possible in real time without moving any information. Data resources don't even have to be in a single repository. The future looks brighter than ever for artificial intelligence and machine learning. Both depend on data to build predictive models that could be used to support a device's potential for autonomy.

Predicting security issues before they happen is another possibility. This is significant, given the vulnerability of the Internet of Things. Big data and relevant programming techniques may keep hackers away from home security systems, automotive electronics, and even toys as cybercriminals seek new ways to obtain data and disrupt lives.

We, therefore, predict programmers will implement artificial intelligence and machine learning to solve some of today's most pervasive computing issues.

Learn Computer Programming with Today's Most Relevant Training Courses

The tools, technologies, and resources available today will have a profound impact on future programming. It's also easier than ever before to obtain training. From the basics of functional programming to specific languages such as Scala, you can learn from the comfort of your own home. For more about the latest online programming courses, contact YoppWorks at 1-888-322-6002 or browse our online training options today.

Information System Entanglement

Information System Entanglement is a problem from a hypothetical conflict. It is a general classification to the problems regarding networks that are negatively affected because of interference. Interference is a conflict that could be sourced back from outside of a network or inside the network. Outside interference is an indirect cause of a negative effect on a series of network. Inside Interference is an indirect cause of a negative effect on a series of network. The series of the network on this theory only refers to two or more connected networks during the interference.

A server is a computer, a device or a program that is dedicated to managing network resources. Servers are often referred to as dedicated because they carry out hardly any other tasks apart

from their server tasks. There are a number of categories of servers, including print servers, file servers, network servers and database servers. In theory, whenever computers share resources with client machines they are considered servers.

Nearly all personal computers are capable of serving as network servers. However, usually, software/hardware system dedicated computers have features and configurations just for this task. For example, dedicated servers may have high-performance RAM, a faster processor, and several high-capacity hard drives. In addition, dedicated servers may be connected to redundant power supplies, several networks, and other servers. Such connection features and configurations are necessary as many client machines and client programs may depend on them to function efficiently, correctly and reliably.

In order to operate in the unique network environment where many computers and hardware/software systems are dependent on just one or several server computers, a server often has special characteristics and capabilities, including the ability to update hardware and software without a restart or reboot, advanced backup capability for frequent backup of critical data, advanced networking performance, automatic (invisible to the user) data transfer between devices, high security for resources, data and memory protection.

Server computers often have special operating systems not usually found on personal computers. Some operating systems are available in both server and desktop versions and use similar interfaces. However, an increase in the reliability of both server hardware and operating systems have blurred the distinctions between desktop and server operating systems.

An application server is a type of server designed to install, operate and host applications and associated services for end users, IT services and organizations. It facilitates the hosting and

delivery of high-end consumer or business applications, which are used by multiple and simultaneously connected local or remote users.

An application server consists of a server operating system (OS) and server hardware that work together to provide computing-intensive operations and services to the residing application. An application server executes and provides user and/or other app access when utilizing the installed application's business/functional logic. Key required features of an application server include data redundancy, high availability, load balancing, user management, data/application security, and a centralized management interface. Moreover, an application server may be connected by enterprise systems, networks or intranet and remotely accessed via the Internet.

Depending on the installed application, an application server may be classified in a variety of ways, including as a Web server, database application server, general purpose application server or enterprise application (EA) server.

The term database server may refer to both hardware and software used to run a database, according to the context. Like software, a database server is the back-end portion of a database application, following the traditional client-server model. This back-end portion is sometimes called the instance. It may also refer to the physical computer used to host the database. When mentioned in this context, the database server is typically a dedicated higher-end computer that hosts the database.

Note that the database server is independent of the database architecture. Relational databases, flat files, non-relational databases: all these architectures can be accommodated on database servers.

In the client-server computing model, there is a dedicated host to run and serve up the resources, typically one or more software applications. There are also several clients who can connect to the server and use the resources offered and hosted by this server.

When considering databases in the client-server model, the database server may be the back-end of the database application (the instance), or it may be the hardware computer that hosts the instance. Sometimes, it may even refer to the combination of both hardware and software.

In smaller and mid-sized setups, the hardware database server will also typically host the server part of the software application that uses the database. If we consider a bank, for instance, the hardware database server will host the software database server and the bank's software application. This application will likely connect to the database via specific ports and use inter-process communication to log into and access the data resident in the database. The users in the bank, seated at their personal computers, will also use the client module of the application installed on their computers to connect to the database. In this example, there are actually two client-server models we are looking at: the database and the application.

In larger setups, the volume of transactions may be such that one computer will be unable to handle the load. In this case, the database software will reside on a dedicated computer and the application on another. In this scenario, there is a dedicated database server, which is the combination of the hardware and software, and a separate dedicated application server.

A network server is a computer designed to act as a central repository and help in providing various resources like hardware access, disk space, printer access, etc., to other computers in the network. A network server might not differ from a workstation in hardware, but the functionality it performs clearly differentiates it from other workstations. Network servers help in simplifying

the different tasks for system administrators including those centering around management. Any configuration or security updates can be applied to a network server instead of individually passing to different computers connected to the network.

References

"8 Predictions of Future Programming Based on 2017 Trends." [Online]. Available: http://yoppworks.com/8-predictions-of-future-programming-based-on-2017-trends/. [Accessed: 26-Apr-2019]

"What is a Server? - Definition from Techopedia," *Techopedia.com*. [Online]. Available: https://www.techopedia.com/definition/2282/server. [Accessed: 26-Apr-2019]